Metrol. Meas. Syst., Vol. 28 (2021) No. 4, pp. 673–691
DOI: 10.24425/mms.2021.137702

PAN POLSKA AKADEMIA NAUK

METROLOGY AND MEASUREMENT SYSTEMS
Index 330930, ISSN 0860-8229
www.metrology.wat.edu.pl

M&MS

A NEW PREDICTIVE FILTER FOR NONLINEAR ALIGNMENT MODEL OF STATIONARY MEMS INERTIAL SENSORS

Hassan Majed Alhassan, Nemat Allah Ghahremani

Malek Ashtar University of Technology, Faculty of Electrical & Computer Engineering, Tehran 15875-1774, Iran
(*hassan.majed.alhassan@gmail.com*, ✉ *ghahremani@mut.ac.ir*, +98 21 22 986 292)

Abstract

This paper proposes a new approach called the Predictive Kalman Filter (PKF) which predicts and compensates model errors of inertial sensors to improve the accuracy of static alignment without the use of external assistance. The uncertain model error is the main problem in the field as the *Micro Electro Mechanical System* (MEMS) inertial sensors have bias which change over time, and these errors are not all observable. The proposed filter determines an optimal equivalent model error by minimizing a quadratic penalty function without augmenting the system state space. The optimization procedure enables the filter to decrease both model uncertainty and external disturbances. The paper first presents the complete formulation of the proposed filter. Then, a nonlinear alignment model with a large misalignment angle is considered. Experimental results demonstrate that the new method improves the accuracy and rapidness of the alignment process as the convergence time is reduced from 550 s to 50 s, and the azimuth misalignment angle correctness is decreased from $52'' \pm 47''$ to $4'' \pm 0.02''$.

Keywords: predictive filter, nonlinear alignment, model error, optimization, MEMS inertial sensors.

© 2021 Polish Academy of Sciences. All rights reserved

1. Introduction

The initial alignment method is the process of determination of initial orientation angles between stationary *Strap-down inertial navigation systems* (SINS) fixed to a carrier and the navigation frame. The precision of initial alignment guarantees an excellent starting status for the navigation phase. Traditionally, the alignment process is divided in two stages; coarse alignment and fine alignment [1]. After the analytic coarse alignment, if the misalignment is small, the error equation of the SINS can be approximated as a linear equation and the precise alignment can be completed using the *Extended Kalman Filter* (EKF) to estimate these small misalignment angles [2]. Unfortunately, due to big random drifts of the gyroscopes in *Micro Electro Mechanical System* (MEMS) inertial sensors, the misalignment angles become too large. Thus, the assumption that the angles are small cannot accurately describe the error statics in the linear alignment

Copyright © 2021. The Author(s). This is an open-access article distributed under the terms of the Creative Commons Attribution-NonCommercial-NoDerivatives License (CC BY-NC-ND 4.0 https://creativecommons.org/licenses/by-nc-nd/4.0/), which permits use, distribution, and reproduction in any medium, provided that the article is properly cited, the use is non-commercial, and no modifications or adaptations are made.

Article history: received April 21, 2021; revised July 1, 2021; accepted July 6, 2021; available online August 22, 2021.

model [3]. Hence, it is important to investigate the nonlinear alignment representation when the misalignment angles are not small. However, the EKF, which is used for nonlinear system state estimation, needs to recognize noise statistics (mean and variance). On other hand, noise statistics, in most applications, are unknown. Further, the EKF resolves the nonlinear system state by converting the nonlinear system to a linear system. Therefore, the linearization error will decrease the filtering performance [4].

Several nonlinear alignment models and filtering algorithms are investigated in the literature to estimate the large heading angle error of SINS. A robust multiplicative quaternion Kalman filter [5] is applied for increasing the reliability of the SINS alignment process, the filter is robust to sensor error uncertainties. Nevertheless, it requires zones of uncertainty and covariance of sensor errors and it does not estimate the deterministic sensor errors. To recover the initial orientation of SINS, [6] proposes a *Cubature Kalman filter* (CKF) with multiple fading factors to correct the process and the measurement noise covariance. A similar CKF approach with a nonlinear alignment model was used in [7] to estimate the in-motion alignment angles. A comparable method is recommended in [8] for SINS alignment on a swing base. [9] proposes a filter based on an *unscented Kalman filter* (UKF) to handle the uncertain noise during determining the initial misalignment angles. However, the computational cost of this method is higher than in the other nonlinear filters. Moreover, the adaptive *unscented particle filter* (UPF) is adopted in [10]. The UPF can decrease the impact of model errors and the uncertainty of the sensor noises to improve the accuracy of the alignment but the computational load of this method is high. [11] reports a smoother filter to enhance the estimation of SINS misalignment angles; however, the filter requires stored data to achieve smoothed attitudes. The estimation of initial large attitudes of SINS in the case of an inaccurate model and non-Gaussian noises can be improved by using external measurements. For example, to enhance the alignment accuracy, a nonlinear alignment model is used in [12] for SINS with a large heading misalignment angle coupled with measurements from a *Global Positioning System* (GPS). An alignment algorithm, based on multiple algorithms which run simultaneously, is proposed in [13] for a complex operating environment. Nevertheless, in the articles above, no details were provided as to the initial alignment of stationary SINS without any external aids. An interesting study is [14] where a gradient descent optimization method is used to overcome the disturbances of the inertial sensor by optimizing a cost function for the initial alignment of a stationary SINS without using the latitude information.

This paper proposes a new *predictive Kalman filter* (PKF) for the nonlinear alignment model of stationary SINS. The concept of the PKF strategy comes from the theory of *predictive control* (PC) [15]. The filter is formulated as a real-time optimization process which automatically predicts equivalent model error by minimizing a quadratic penalty function. The equivalent model error covers both the unknown time-varying statistics and the modelling errors such as linearization errors. In other words, this equivalent model error describes the uncertainty of SINS model errors including bias, scale factors, and misalignment angles of the inertial sensors. Consequently, the equivalent model error is estimated to compensate for the inertial sensor errors. The optimization procedure enables the PKF to decrease model uncertainty and external disturbances. Therefore, this method improves the navigation alignment process.

The paper is organized as follows. Section 2 introduces the problem statement. Then, the formulation details of the novel predictive Kalman filtering are addressed in Section 3. Section 4 presents the nonlinear alignment model; nonlinear attitude equations, velocity error equations, and the measurement model. Section 5 explains the experimental results. Finally, Section 6 summarizes the main outcomes of the research and provides the conclusions.

2. Problem definition

In real-life situations, if the azimuth misalignment angle is large, then the system becomes highly nonlinear. Due to linearization and system model errors, the traditional EKF algorithm does not offer any convergence guarantee. In this paper, to solve the nonlinear alignment model, the following mathematical formula of the discrete-time nonlinear stochastic systems is considered:

$$\begin{cases} \mathbf{x}_{k+1} = f(\mathbf{x}_k, k) + \mathbf{w}_k, \\ \mathbf{y}_{k+1} = \mathbf{H}_{k+1}\mathbf{x}_{k+1} + \mathbf{v}_{k+1}. \end{cases} \tag{1}$$

In (1), k describes the step time, $\mathbf{x}_k$ is the state vector of a nonlinear system model $f(\mathbf{x}_k, k)$, the output vector $\mathbf{y}_{k+1}$ and the measurement matrix $\mathbf{H}_{k+1}$ describe the linear measurement equation. The system noise $\mathbf{w}_k$ and measurement noise $\mathbf{v}_{k+1}$ are uncorrelated, zero-mean, Gaussian white noise with unknown covariance matrices $\mathbf{Q}_k$ and $\mathbf{R}_k$, respectively. The first approximation of the Taylor expansion around the optimal state estimate is used to convert (1) to a linear equation as follows:

$$f(\mathbf{x}_k, k) = f(\hat{\mathbf{x}}_k, k) + \frac{\partial f(\hat{\mathbf{x}}_k, k)}{\partial \hat{\mathbf{x}}_k}[\mathbf{x}_k - \hat{\mathbf{x}}_k] + \text{H.O.T.} \tag{2}$$

The term H.O.T. represents the higher-order terms in the expansion. Denoting the Jacobian matrix $\dfrac{\partial f(\hat{\mathbf{x}}_k, k)}{\partial \hat{\mathbf{x}}_k}$ as $\mathbf{F}_k$ and substituting (2) into (1) we obtain the following linear model:

$$\begin{cases} \mathbf{x}_{k+1} = \mathbf{F}_k\mathbf{x}_k + \Delta\mathbf{L}_k + \mathbf{w}_k, \\ \Delta\mathbf{L}_k = f(\hat{\mathbf{x}}_k, k) - \mathbf{F}_k\hat{\mathbf{x}}_k + \text{H.O.T.} \end{cases} \tag{3}$$

The term $\Delta\mathbf{L}_k$ contains the linearization error and the unknown high order in the Taylor expansion. We supposing that (3) is a system with unknown model error $\Delta\mathbf{F}_k$. Therefore, even if the noise statistics are known, the model error of the system will make the traditional Kalman filter performance worse. However, a linear model can replace the nonlinear model (1). It includes an equivalent model error $\mathbf{m}_k$ and has the following form:

$$\begin{aligned} \mathbf{x}_{k+1} &= (\mathbf{F}_k + \Delta\mathbf{F}_k)\mathbf{x}_k + \Delta\mathbf{L}_k + \mathbf{w}_k \\ &= \mathbf{F}_k\mathbf{x}_k + \mathbf{m}_k + \mathbf{w}_k. \end{aligned} \tag{4}$$

The following relation formulates the unknown equivalent model error $\mathbf{m}_k$:

$$\mathbf{m}_k = \Delta\mathbf{F}_k\mathbf{x}_k + \Delta\mathbf{L}_k = \Delta\mathbf{F}_k\mathbf{x}_k + f(\hat{\mathbf{x}}_k, k) - \mathbf{F}_k\hat{\mathbf{x}}_k + \text{H.O.T.} \tag{5}$$

The equivalent model error $\mathbf{m}_k$ covers the unknown process noise statistics and the system model error. The SINS model error includes bias, scale factor, and misalignment angles of MEMS inertial sensors (the gyroscopes and accelerometers). Therefore, the filtering problem of the real nonlinear system (1) is transformed to a linear system with unknown time varying noise which must be identified and compensated for. To solve this problem, the presented PKF filter is designed for predicting the future values (in finite horizon p) of the unknown equivalent model error using a quadratic penalty function which minimizes the weighted squares of the measurement residual term and the model error increment term as follows:

$$\begin{aligned} &\mathbf{J}_{k+1}(\mathbf{x}_k, \Delta\mathbf{m}_k) = \min_{\Delta\mathbf{m}_{k,p}} \left(\left\| \hat{\mathbf{y}}_{k+p} - \mathbf{y}^r_{k+p} \right\|^2_{\mathbf{R}_e} + \left\| \Delta\mathbf{m}_{k+p-1} \right\|^2_{\mathbf{Q}_m} \right) \\ &\text{subject to: } \mathbf{x}_{k+1} = f(\mathbf{x}_k, \mathbf{m}_k) + \mathbf{w}_k, \quad \mathbf{y}_{k+1} = \mathbf{H}\,\mathbf{x}_{k+1} + \mathbf{v}_{k+1} \quad \text{and} \quad \mathbf{m}_k = \mathbf{m}_{k-1} + \Delta m_k. \end{aligned} \tag{6}$$

The penalty function optimizes the unknown increment model error $\Delta\mathbf{m}$ at each step. The following sections describe the principle of the PKF filter.

3. Predictive Kalman Filter

This section will present the fundamental background of the PKF necessary for the discussion of this paper. We will start with the derivation steps of the subspace of the prediction equation which forms the core of the PKF approach. Next, the formulation of the model error and PKF equations are presented.

The Predictive Kalman Filter is a new real-time estimation approach based on the optimization problem. The filter applies a quadratic penalty function to predict the future values of an equivalent model error which optimizes the future output of the system at each sampling time. The penalty function penalizes both the measurement error and the model error. After that, the designed filter compensates for the model error in the state space equations and corrects the noised measurements. The prediction part in Fig. 1 explains the methodology for predicting the future of the optimal equivalent model error within a finite horizon.

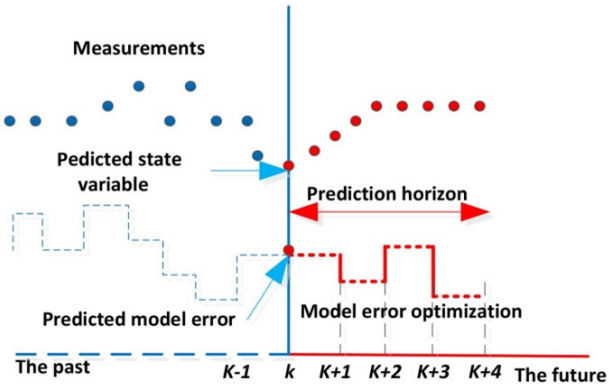


Fig. 1. Optimization methodology for model error.

After the optimization process, only the first element of the predicted model error is considered. In every time step, the prediction horizon is moved and the optimization process is repeated. This iterative optimization procedure compensates for the equivalent error of the filter model.

3.1. Subspace of prediction

The discrete-time state-space (4) for a *linear time-variant* (LTV) system is described in the incremental form for $t = k + 1$ as follows:

$$\begin{cases} \Delta\hat{\mathbf{x}}_{k+1/k} &= \mathbf{F}_k \Delta\hat{\mathbf{x}}_k + \Delta\hat{\mathbf{m}}_k \,, \\ \hat{\mathbf{x}}_{k+1/k} &= \mathbf{H}\Delta\hat{\mathbf{x}}_{k+1/k} \,. \end{cases} \tag{7}$$

In (7), the terms $\Delta\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_k - \hat{\mathbf{x}}_{k-1}$ and $\Delta\hat{\mathbf{m}}_k = \hat{\mathbf{m}}_k - \hat{\mathbf{m}}_{k-1}$ are the incremental form of the states and the model error. Supposing the sampling time is small and the system matrix $\mathbf{F}_k$ is invariant during the prediction horizon, then, at time $t = k+2$, the state space (7) can be rewritten as follows:

$$\Delta\hat{\mathbf{x}}_{k+2/k} = \mathbf{F}_k \Delta\hat{\mathbf{x}}_{k+1/k} + \Delta\hat{\mathbf{m}}_{k_{k+1/k}} \,. \tag{8}$$

By substituting (7) in (8) to eliminate $\hat{\mathbf{x}}_{k+1/k}$, the following relation is obtained:

$$\begin{cases} \Delta\hat{\mathbf{x}}_{k+2/k} = \mathbf{F}_k^2\Delta\hat{\mathbf{x}}_k + \mathbf{F}_k\Delta\hat{\mathbf{m}}_k + \Delta\hat{\mathbf{m}}_{k+1/k}\,, \\ \hat{\mathbf{x}}_{k+2/k} = \mathbf{H}\mathbf{F}_k^2\Delta\hat{\mathbf{x}}_k + \mathbf{H}\mathbf{F}_k\Delta\hat{\mathbf{m}}_k + \mathbf{H}\Delta\hat{\mathbf{m}}_{k+1/k}\,. \end{cases} \tag{9}$$

In step time $t = k + p$, the predicted state variables and the output take the following form:

$$\begin{cases} \Delta\hat{\mathbf{x}}_{k+p/k} = \mathbf{F}_k^p\Delta\hat{\mathbf{x}}_k + \displaystyle\sum_{i=1}^{p}\mathbf{F}_k^{p-i}\Delta\hat{\mathbf{m}}_{k+i-1}\,, \\ \hat{\mathbf{x}}_{k+p/k} = \mathbf{H}\mathbf{F}_k^p\Delta\hat{\mathbf{x}}_k + \displaystyle\sum_{i=1}^{p}\mathbf{H}\mathbf{F}_k^{p-i}\Delta\hat{\mathbf{m}}_{k+i-1}\,. \end{cases} \tag{10}$$

Collecting the previous prediction results for $t \in \{k+1 \cdots k+p\}$ will give the following formulation for the subspace of prediction:

$$\begin{cases} \mathbf{X}_{k+1,p} = \mathbf{A}_p\Delta\hat{\mathbf{x}}_k + \mathbf{B}_p\,\mathbf{M}_{k,p-1}\,, \\ \mathbf{Y}_{k+1,p} = \mathbf{C}_p\Delta\hat{\mathbf{x}}_k + \mathbf{D}_p\,\mathbf{M}_{k,p-1}\,. \end{cases} \tag{11}$$

Where the vectors and the matrices are defined according to the following relations:

$$\begin{cases} \mathbf{X}_{k+1,p} = \left[\Delta\hat{\mathbf{x}}_{k+1/k}\ \ \Delta\hat{\mathbf{x}}_{k+2/k}\ \ \ldots\ \Delta\hat{\mathbf{x}}_{k+p/k}\right]^T_{(np\times 1)}, \\ \mathbf{Y}_{k+1,p} = \left[\hat{\mathbf{x}}_{k+1/k}\ \ \hat{\mathbf{x}}_{k+2/k}\ \ \ldots\ \hat{\mathbf{x}}_{k+p/k}\right]^T_{(mp\times 1)}, \quad \mathbf{M}_{k,p-1} = \left[\Delta\hat{\mathbf{m}}_k\ \ \Delta\hat{\mathbf{m}}_{k+1}\ \ \ldots\ \Delta\hat{\mathbf{m}}_{k+p-1}\right]^T_{(np\times 1)}, \\ \mathbf{A}_p = \left[\mathbf{F}_k\ \ \mathbf{F}_k^2\ \ \mathbf{F}_k^3\ \ \ldots\ \mathbf{F}_k^p\right]^T_{(np\times n)}, \quad \mathbf{C}_p = \left[\mathbf{H}\mathbf{F}_k\ \ \mathbf{H}\mathbf{F}_k^2\ \ \mathbf{H}\mathbf{F}_k^3\ \ \ldots\ \mathbf{H}\mathbf{F}_k^p\right]^T_{(mp\times n)}, \\ \mathbf{B}_p = \begin{bmatrix} \mathbf{I}_{n\times n} & \mathbf{0} & \mathbf{0} & \ldots & \mathbf{0} \\ \mathbf{F}_k & \mathbf{I}_{n\times n} & \mathbf{0} & \ldots & \mathbf{0} \\ \mathbf{F}_k^2 & \mathbf{F}_k & \mathbf{I}_{n\times n} & \ldots & \mathbf{0} \\ \vdots & \vdots & \vdots & \ldots & \mathbf{0} \\ \mathbf{F}_k^{p-1} & \mathbf{F}_k^{p-2} & \mathbf{F}_k^{p-3} & \ldots & \mathbf{I}_{n\times n} \end{bmatrix}_{(np\times np)}, \quad \mathbf{D}_p = \begin{bmatrix} \mathbf{H} & \mathbf{0} & \mathbf{0} & \ldots & \mathbf{0} \\ \mathbf{H}\mathbf{F}_k & \mathbf{H} & \mathbf{0} & \ldots & \mathbf{0} \\ \mathbf{H}\mathbf{F}_k^2 & \mathbf{H}\mathbf{F}_k & \mathbf{H} & \ldots & \mathbf{0} \\ \vdots & \vdots & \vdots & \ldots & \mathbf{0} \\ \mathbf{H}\mathbf{F}_k^{p-1} & \mathbf{H}\mathbf{F}_k^{p-2} & \mathbf{H}\mathbf{F}_k^{p-3} & \ldots & \mathbf{H} \end{bmatrix}_{(mp\times np)}. \end{cases} \tag{12}$$

In (11), the future output of the system is predicted in terms of past state variables increment and future model error increment values. The coefficients of the matrices of this subspace ($\mathbf{A}_p$, $\mathbf{B}_p$, $\mathbf{C}_p$ and $\mathbf{D}_p$) are obtained directly from the data at $t = k$.

3.2. Predictive filter algorithm

The predictive algorithm requires a penalty function in its optimization process. For the SINS self-alignment, the penalty function penalizes the weighted norm of the current innovation measurement states and the norm of the model error according to the following form.

$$\mathbf{J}_{k+1} = \left(\hat{\mathbf{Y}}_{k+1,p} - \mathbf{r}\right)^T \mathbf{R}_e \left(\hat{\mathbf{Y}}_{k+1,p} - \mathbf{r}\right) + \Delta\mathbf{M}_{k,p-1}^T\mathbf{Q}_m\Delta\mathbf{M}_{k,p-1}\,. \tag{13}$$

The second term in the above penalty function is added to reduce the cost of model error. The measurement equation contains the pseudo horizontal velocity measurements $\delta v_N = \delta v_E = 0$

used as a direct measure of the errors, in this case $\mathbf{r} = \mathbf{0}$. The diagonal positive definite matrices $\mathbf{R}_e$ and $\mathbf{Q}_m$ are the weighting matrices of measurement error and model error increment, respectively. These matrices are the tuning parameters of the filter. After some calculations, the penalty function can be expressed as follows:

$$\begin{aligned}\mathbf{J}_{k+1} = {} & \Delta\mathbf{M}_{k,p-1}^{T}\left[\mathbf{D}_p^T\mathbf{C}_p^T\mathbf{R}_e\mathbf{C}_p\mathbf{D}_p + \mathbf{Q}_m\right]\Delta\mathbf{M}_{k,p-1} \\ & + 2\left[\left(\mathbf{C}_p\mathbf{A}_p\Delta\hat{\mathbf{x}}_k\right)^T\mathbf{R}_e\mathbf{C}_p\mathbf{D}_p\right]\Delta\mathbf{M}_{k,p-1} + \left(\mathbf{C}_p\mathbf{A}_p\Delta\hat{\mathbf{x}}_k\right)^T\mathbf{R}_e\left(\mathbf{C}_p\mathbf{A}_p\Delta\hat{\mathbf{x}}_k\right). \end{aligned} \tag{14}$$

The optimization problem is solved by minimization of the penalty function using the least-squares method, which allows analytical solution for an unconstrained problem and penalizes larger errors more than smaller errors. After differentiating the penalty function $\mathbf{J}_{k+1}$ for $\Delta\mathbf{M}_{k,p-1}$, the optimal solution $\Delta\mathbf{M}^*_{k,p-1}$ is as follows:

$$\Delta\mathbf{M}^*_{k,p-1} = \mathbf{K}_{PF}\Delta\hat{\mathbf{x}}_k\,, \tag{15}$$

where the algorithm gain equals:

$$\mathbf{K}_{PF} = -\left[\mathbf{C}_p^T\mathbf{D}_p^T\mathbf{R}_e\mathbf{D}_p\mathbf{C}_p + \mathbf{Q}_m\right]^{-1}\mathbf{C}_p^T\mathbf{D}_p^T\mathbf{R}_e\mathbf{C}_p\mathbf{A}_p\,. \tag{16}$$

Since optimization takes place at each sampling interval, only the first elements $\Delta\mathbf{m}^*_k$ of the sequence $\Delta\mathbf{M}^*_{k,p-1}$ are considered, thus:

$$\begin{cases} \Delta\mathbf{m}^*_k = -\left[\mathbf{I}_{n\times n} \quad 0 \quad \ldots \quad 0\right]^T\mathbf{K}_{PF}\Delta\hat{\mathbf{x}}_k\,, \\ \mathbf{m}_k = \mathbf{m}_{k-1} + \Delta\mathbf{m}^*_k\,. \end{cases} \tag{17}$$

At the next time step, a new output is measured and, again, a new optimal model error sequence $\Delta\mathbf{m}^*_k$ is applied to the system. Then, the computed model error $\mathbf{m}_k$ is used to compensate for the output measurement (in this case is the bias of the inertial sensors). This method provides more compatibility for the estimation of the state space variables, due to the existence of tuning parameters, which are the weighting matrices ($\mathbf{R}_e$ and $\mathbf{Q}_m$) and the prediction horizon p. By adjusting these parameters the specific stability and performance can be guaranteed for the system. Fig. 2, shows the principle of the proposed approach for the alignment process.

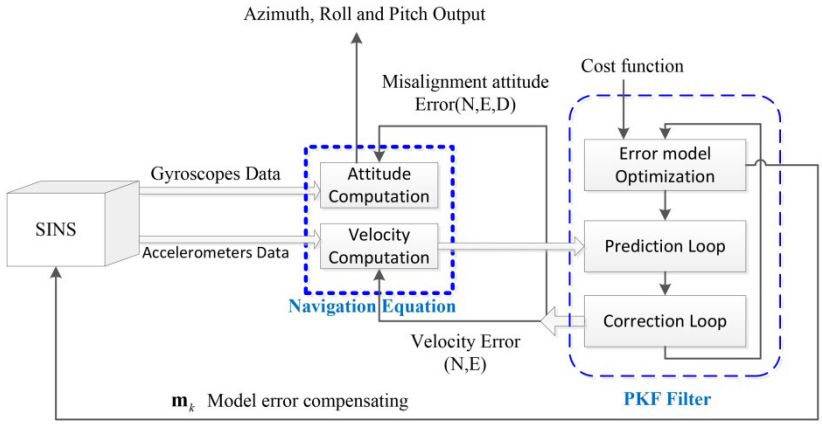


Fig. 2. The principle of the PKF approach for the alignment process.

Metrol. Meas. Syst., Vol. 28 (2021), No. 4, pp. 673–691
DOI: 10.24425/mms.2021.137702

4. Nonlinear alignment model

The precise alignment mode is chosen to determine the azimuth error through purely inertial measurements. Unfortunately, the linear alignment model is offered for small angles only. But, in most situations, the initial orientation errors are very large. Therefore, it is necessary to consider a nonlinear alignment model. In this section, a nonlinear approach is presented which does not require rough alignment, and the azimuth misalignment is assumed to be large. The PKF filter algorithm is used for initial attitude estimation. The simplified PKF algorithm is illustrated in Fig. 3.

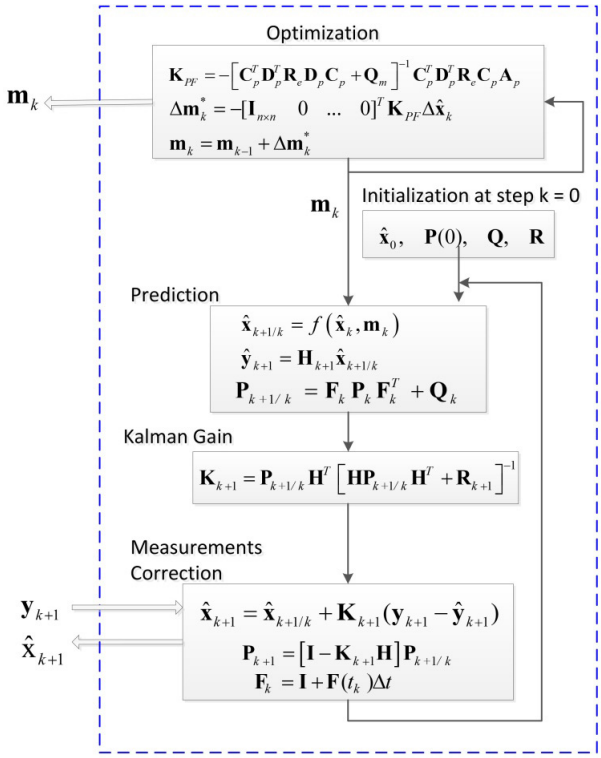


Fig. 3. PKF algorithm.

4.1. Nonlinear attitude error equation

The alignment process for a SINS determines the direction cosine matrix (DCM) or the matrix $\mathbf{C}_b^n$, The SINS measurements in the body frame are resolved into the navigation NED frame (the geographic north, east and down coordinate system). The true $\mathbf{C}_b^n$ is defined in the NED frame as:

$$\mathbf{C}_b^n = \begin{bmatrix} c\theta\, s\psi & s\phi\, s\theta\, c\psi - c\phi\, s\psi & c\phi\, s\theta\, c\psi + s\phi\, s\psi \\ c\theta\, s\psi & s\phi\, s\theta\, s\psi + c\phi\, c\psi & c\phi\, s\theta\, s\psi - s\phi\, c\psi \\ -s\theta & s\phi\, c\theta & c\phi\, c\theta \end{bmatrix}, \tag{18}$$

where c and s denote the cosine and the sine functions. θ, ϕ and ψ are the pitch, roll and azimuth angles, respectively.

The derivative of the transformation matrix can be written according to the following equation:

$$\dot{\mathbf{C}}_b^n = \mathbf{C}_b^n \boldsymbol{\Omega}_{nb}^b = \mathbf{C}_b^n \boldsymbol{\Omega}_{ib}^b - \boldsymbol{\Omega}_{in}^n \mathbf{C}_b^n , \tag{19}$$

where the matrix $\boldsymbol{\Omega}_{ib}^b$ represents the skew-symmetric matrix form of $\boldsymbol{\omega}_{ib}^b$; the true angular velocity of the body frame relative to an inertial frame. The matrix $\boldsymbol{\Omega}_{in}^n$ represents the skew-symmetric matrix of $\boldsymbol{\omega}_{in}^n$; the true angular velocity of the navigation frame relative to an inertial frame. The IMU measurements mixed with errors lead to a transformation matrix $\mathbf{C}_b^{n'}$, where the n'-frame is assumed to be the computed version of the true navigation NED frame. The difference between n'-frame and n-frame is the initial misalignment error which needs to be compensated for by the alignment process. The derivative of the transformation matrix $\mathbf{C}_b^{n'}$ is written according to the following equation:

$$\dot{\mathbf{C}}_b^{n'} = \mathbf{C}_b^{n'} \hat{\boldsymbol{\Omega}}_{nb}^b = \mathbf{C}_b^{n'} \hat{\boldsymbol{\Omega}}_{ib}^b - \hat{\boldsymbol{\Omega}}_{in}^{n'} \mathbf{C}_b^n . \tag{20}$$

The output of the gyroscopes or the measured rate $\hat{\boldsymbol{\omega}}_{ib}^b$ is related to the true rate $\boldsymbol{\omega}_{ib}^b$ by the following equation:

$$\hat{\boldsymbol{\omega}}_{ib}^b = \boldsymbol{\omega}_{ib}^b + \delta\boldsymbol{\omega}_{ib}^b . \tag{21}$$

The error $\delta\boldsymbol{\omega}_{ib}^b$ contains all drift errors of the gyroscopes. A new transformation matrix $\mathbf{C}_{n'}^n$ can be defined to express the difference between n'-frame and n-frame as follows:

$$\mathbf{C}_b^n = \mathbf{C}_{n'}^n \mathbf{C}_b^{n'} . \tag{22}$$

Assuming that the horizontal angles are small and there is a large angle in azimuth between n'-frame and n-frame, the matrix $\mathbf{C}_{n'}^n$ is:

$$\mathbf{C}_{n'}^n = \begin{bmatrix} c\varphi_D & -s\varphi_D & \varphi_E c\varphi_D + \varphi_N s\varphi_D \\ s\varphi_D & c\varphi_D & \varphi_E s\varphi_D - \varphi_N c\varphi_D \\ -\varphi_E & \varphi_N & 1 \end{bmatrix}, \tag{23}$$

where φ_N and φ_E are the small horizontal misalignment angles of roll and pitch, respectively, φ_D is the large azimuth angle. After some rearrangement (see Appendix A), the SINS misalignment angles satisfy the following equation:

$$\dot{\boldsymbol{\varphi}} = \left(\mathbf{I} - \mathbf{C}_n^{n'}\right) \boldsymbol{\omega}_{in}^n + \delta\boldsymbol{\omega}_{in}^n - \mathbf{C}_b^{n'} \delta\boldsymbol{\omega}_{ib}^b . \tag{24}$$

Vector $\boldsymbol{\varphi} = \begin{bmatrix} \varphi_N & \varphi_E & \varphi_D \end{bmatrix}^T$ denotes the mathematical SINS misalignment angles expressed in n'-frame. In the stationary alignment, the position is known and fixed. Therefore, the value of angular velocity of n-frame relative to earth frame $\boldsymbol{\omega}_{en}^n$ being equal to zero, then:

$$\begin{cases} \boldsymbol{\omega}_{in}^n = \boldsymbol{\omega}_{ie}^n + \boldsymbol{\omega}_{en}^n = \boldsymbol{\omega}_{ie}^n = \begin{bmatrix} \omega_e \cos\varphi & 0 & -\omega_e \sin\varphi \end{bmatrix}^T , \\ \delta\boldsymbol{\omega}_{in}^n = \begin{bmatrix} \dfrac{\delta v_E}{r}, & \dfrac{-\delta v_N}{r}, & \dfrac{-\delta v_E}{r} \tan\varphi \end{bmatrix}^T . \end{cases} \tag{25}$$

Vector $\boldsymbol{\omega}_{in}^n$ is the angular velocity of n-frame relative to the inertial frame, the vector $\delta\boldsymbol{\omega}_{in}^n$ is the perturbation of $\boldsymbol{\omega}_{in}^n$, $\boldsymbol{\omega}_{ie}^n$ is the earth rotation rate relative to inertial frame, $\boldsymbol{\omega}_{en}^n$ is the angular rate of the reference frame relative to the earth frame. Parameter ω_e is the earth's angular velocity, δv_N and δv_E are the north and east components of the velocity error, r is the earth radius and φ is the geodetic latitude. The transformation matrix between the true n-frame and the computed

n'-frame ($\mathbf{C}_n^{n'}$) can be calculated from (23) as $\mathbf{C}_n^{n'} = \left(\mathbf{C}_{n'}^{n}\right)^T$. From (24) and (25) the nonlinear attitude error equations can be described as:

$$\begin{cases} \dot{\varphi}_N = (1 - c\varphi_D)\omega_N + \varphi_E\omega_D + \delta v_E^{n'}/r - \Delta\omega_N\,, \\ \dot{\varphi}_E = \omega_N s\varphi_D - \varphi_x\omega_D - \delta v_N^{n'}/r - \Delta\omega_E\,, \\ \dot{\varphi}_D = -(\varphi_E c\varphi_D + \varphi_N s\varphi_D)\omega_N - \delta v_E \tan\varphi/r - \Delta\omega_D\,. \end{cases} \tag{26}$$

In (26), we denoted $\omega_D = -\omega_e \sin\varphi$ and $\omega_N = \omega_e \cos\varphi$ for simplicity, the symbols c and s denote the cosine and the sine functions, $\Delta\omega_N$, $\Delta\omega_E$ and $\Delta\omega_D$ are the components of gyro bias expressed in n'-frame and they are equal to the components of the vector $\mathbf{C}_b^{n'}\delta\boldsymbol{\omega}_{ib}^b$ (see (24)), where vector $\delta\boldsymbol{\omega}_{ib}^b$ denotes the gyroscopes bias.

4.2. Nonlinear Velocity error equation

In the case of static alignment, the SINS position remains constant and there is no velocity drift direction, the expression of the navigation equation in the NED frame can be written as follows [1, 16]:

$$\dot{\mathbf{v}}_e^n = \mathbf{C}_b^n\mathbf{f}^b - \left(2\boldsymbol{\omega}_{ie}^n + \boldsymbol{\omega}_{en}^n\right) \times \mathbf{v}_e^n + \mathbf{g}^n, \tag{27}$$

where vector $\mathbf{f}^b$ is the true specific force vector, $\mathbf{C}_b^n$ denotes the attitude matrix from the body frame to the n-frame, $\mathbf{v}_e^n$ is the velocity of n-frame relative to the earth, and $\mathbf{g}^n$ is the gravity vector. For stationary SINS, $\boldsymbol{\omega}_{en}^n$ is equal to zero, therefore, (27) is rewritten as follows:

$$\dot{\mathbf{v}}_e^n = \mathbf{C}_b^n\mathbf{f}^b - 2\boldsymbol{\omega}_{ie}^n \times \mathbf{v}_e^n + \mathbf{g}^n, \tag{28}$$

Based on the previous equation, the measured velocity in n'-frame can be written as:

$$\dot{\mathbf{v}}_e^{n'} = \mathbf{C}_b^{n'}\hat{\mathbf{f}}^b - 2\hat{\boldsymbol{\omega}}_{ie}^n \times \mathbf{v}_e^{n'} + \mathbf{g}^{n'}, \tag{29}$$

where vector $\hat{\mathbf{f}}^b$ is the specific force vector measured by the accelerometers, the measured $\hat{\mathbf{f}}^b$ is related to the true $\mathbf{f}^b$ by $\hat{\mathbf{f}}^b = \mathbf{f}^b + \delta\mathbf{f}^b$. Vector $\delta\mathbf{f}^b$ contains all bias errors of the accelerometers, $\mathbf{C}_b^{n'}$ denotes the transformation matrix from the body frame to n'-frame; $\mathbf{C}_b^{n'} = \mathbf{C}_n^{n'}\mathbf{C}_b^n$, and vector $\mathbf{v}_e^{n'}$ the velocity of n'-frame relative to the earth frame. By subtracting (28) from (29) and assuming $\mathbf{g}^{n'} = \mathbf{g}^n$ (small horizontal misalignment angles), the velocity differential equation of SINS is:

$$\delta\dot{\mathbf{v}}^{n'} = \dot{\mathbf{v}}_e^{n'} - \dot{\mathbf{v}}_e^n = \left(\mathbf{C}_n^{n'} - \mathbf{I}\right)\mathbf{C}_b^n\mathbf{f}^b - 2\boldsymbol{\omega}_{ie}^n \times \delta\mathbf{v}^{n'} + \mathbf{C}_b^{n'}\delta\mathbf{f}^b. \tag{30}$$

Equation (30) represents the nonlinear velocity error model of SINS. With no vertical deflection error ($\delta v_D^{n'} = 0$) and zero nominal velocity ($v_D^{n'} = 0$) thus (30) is:

$$\begin{cases} \delta\dot{v}_N^{n'} = (1 - c\varphi_D)f_N^{n'} + s\varphi_D f_E^{n'} + 2\omega_D\delta v_E^{n'} - (\varphi_E c\varphi_D + \varphi_N s\varphi_D)f_D^{n'} + \Delta f_N\,, \\ \delta\dot{v}_E^{n'} = -s\varphi_D f_N^{n'} + (1 - c\varphi_D)f_E^{n'} - 2\omega_D\delta v_N^{n'} + (-\varphi_E s\varphi_D + \varphi_N c\varphi_D)f_D^{n'} + \Delta f_E\,. \end{cases} \tag{31}$$

In (31) $f_N^{n'}$, $f_E^{n'}$ and $f_D^{n'}$ denote the components of $\mathbf{f}^b$ transformed to n'-frame, Δf_N and Δf_E are the north and east accelerometer bias expressed in n'-frame and they are equal to the first two components of vector $\mathbf{C}_b^{n'}\delta\mathbf{f}^b$ (see (30)).

4.3. Measurement model

In this paper model error is included in the process model as correction to compensate for additive systematic error. The unknown value of the correction can be determined with the PKF algorithm. The correction uncertainty contributes to the uncertainty associated with the estimated variables and they are quantified by covariance or standard deviation.

When the vehicle is static, the zero-velocity information in navigation frame is used as external reference and the measurement vector can be constructed as $\mathbf{y}_t = \begin{bmatrix} v_N^{n'} - 0 & v_E^{n'} - 0 \end{bmatrix}^T = \begin{bmatrix} \delta v_N^{n'} & \delta v_E^{n'} \end{bmatrix}^T$, where $\delta v_N^{n'}$ and $\delta v_E^{n'}$ are the measurement variables equal to $v_N^{n'}$ and $v_E^{n'}$, respectively. In the alignment mode, the SINS is stationary relative to the earth and the velocity components should be zero, any external motion will be treated as measurement noise. The linear relation between observations and states is given by:

$$\begin{aligned} \mathbf{y}_t &= \mathbf{H}\mathbf{x}_t + \mathbf{v}_t\,, \\ \mathbf{H} &= \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix}, \end{aligned} \tag{32}$$

where: $\mathbf{v}_t$ is the white measurement noise with covariance $\mathbf{R}$ (This input quantity value and uncertainty is brought into the measurement from the datasheet of the sensors), matrix $\mathbf{H}$ is the relationship between the measured vector and state vector $\mathbf{x}_t = \begin{bmatrix} \delta v_N^{n'} & \delta v_N^{n'} & \varphi_N & \varphi_E & \varphi_D \end{bmatrix}^T$.

4.4. Nonlinear alignment model for the PKF algorithm

From (26), (31) and (32), the continuous nonlinear alignment model is formulated as:

$$\begin{aligned} \dot{\mathbf{x}}_{t+1} &= f\left(\mathbf{x}_t,\, \mathbf{m}'_t\right) + \mathbf{w}_t\,, \\ \mathbf{y}_{t+1} &= \mathbf{H}\mathbf{x}_{t+1} + \mathbf{v}_{t+1}\,. \end{aligned} \tag{33}$$

In (33) t describes the step time. The state variables needed to define in n'-frame are the horizontal velocity errors and the orientation errors (misalignment angles of roll, pitch and yaw). Therefore, the state vector is $\mathbf{x}_t = \begin{bmatrix} \delta v_N^{n'} & \delta v_N^{n'} & \varphi_N & \varphi_E & \varphi_D \end{bmatrix}^T$. Vector $\mathbf{m}'_t$ is the model error (residual bias of gyroscopes and accelerometers transformed to n'-frame). The vector of the model error is defined as: $\mathbf{m}'_t = \begin{bmatrix} \Delta f_N & \Delta f_E & \Delta\omega_N & \Delta\omega_E & \Delta\omega_D \end{bmatrix}^T$, where Δf_N and Δf_E are the north and east components of accelerometer bias. $\Delta\omega_N$, $\Delta\omega_E$ and $\Delta\omega_D$ are the components of gyroscopes drifts. The nonlinear model of the errors is defined as a nonlinear function:

$$f\left(\mathbf{x}_t,\, \mathbf{m}'_t\right):$$

$$\begin{cases} \delta\dot{v}_N^{n'} = (1 - c\varphi_D) f_N^{n'} + s\varphi_D f_E^{n'} + 2\omega_D \delta v_E^{n'} - (\varphi_E c\varphi_D + \varphi_N s\varphi_D) f_D^{n'} + \Delta f_N\,, \\ \delta\dot{v}_E^{n'} = -s\varphi_D f_N^{n'} + (1 - c\varphi_D) f_E^{n'} - 2\omega_D \delta v_N^{n'} + (-\varphi_E s\varphi_D + \varphi_N c\varphi_D) f_D^{n'} + \Delta f_E\,, \\ \dot{\varphi}_N = (1 - c\varphi_D)\omega_N + \varphi_E\omega_D + \delta v_E^{n'}/r - \Delta\omega_N\,, \\ \dot{\varphi}_E = \omega_N s\varphi_D - \varphi_x\omega_D - \delta v_N^{n'}/r - \Delta\omega_E\,, \\ \dot{\varphi}_D = -(\varphi_E c\varphi_D + \varphi_N s\varphi_D)\omega_N - \delta v_E \tan\varphi/r - \Delta\omega_D\,. \end{cases} \tag{34}$$

The Jacobian matrix is given as follows:

$$\mathbf{F}_t = \frac{\partial \mathbf{f}(\mathbf{x}_t)}{\partial \mathbf{x}_t} = \begin{bmatrix} 0 & 2\omega_D & -s\varphi_D f_D^{n'} & -c\varphi_D f_D^{n'} & f_E^{n'} c\varphi_D + f_N^{n'} s\varphi_D + (\varphi_E s\varphi_D - \varphi_N c\varphi_D) f_D^{n'} \\ -2\omega_D & 0 & c\varphi_D f_D^{n'} & -s\varphi_D f_D^{n'} & f_E^{n'} s\varphi_D - f_N^{n} c\varphi_D - (\varphi_E c\varphi_D + \varphi_N s\varphi_D) f_D^{n'} \\ 0 & 1/r & 0 & \omega_D & \omega_N s\varphi_D \\ -1/r & 0 & -\omega_D & 0 & \omega_N c\varphi_D \\ 0 & -\tan\varphi/r & -\omega_N s\varphi_D & -\omega_N c\varphi_D & (-\varphi_N c\varphi_D + \varphi_E s\varphi_D)\omega_N \end{bmatrix}. \tag{35}$$

The discrete time form of the continuous nonlinear alignment model (33) is identical to (1). From the considerations in Section 2, the linearized model includes an equivalent model error $\mathbf{m}_k$. The equivalent model error covers the unknown process noise statistics and the system modelling error, such as linearization as described in (5). Here, $\mathbf{m}_k$ is as follows:

$$\mathbf{m}_k = \mathbf{m}'_t + \Delta\mathbf{L}_k \approx \mathbf{m}'_k + f\left(\hat{\mathbf{x}}_k, \mathbf{m}'_k\right) - \mathbf{F}_k\hat{\mathbf{x}}_k + \text{H.O.T.} \tag{36}$$

The SINS model error includes gyroscope and accelerometer biase projected in n'-frame and the linearization errors. Therefore, the filtering problem of the nonlinear alignment model (33) is transformed to linear system $\mathbf{x}_{k+1} = \mathbf{F}_k\mathbf{x}_k + \mathbf{m}_k + \mathbf{w}_k$, with unknown equivalent modelling error $\mathbf{m}_k$ which must be identified and compensated for. To estimate m_k, first we construct the subspace of prediction in incremental form as $\Delta\hat{\mathbf{x}}_{k+1/k} = \mathbf{F}_k\Delta\hat{\mathbf{x}}_k + \Delta\hat{\mathbf{m}}_k$ (see (7)). Where the discreet system matrix is $\mathbf{F}_k = \mathbf{I} + \mathbf{F}_t\Delta t$; Δt is the sampling time. After forming the vectors and the matrices (12), the PKF algorithm (Section 3.2) calculates the optimal incremental model error $\Delta\mathbf{m}_k^*$. Therefore, the unknown equivalent model error $\mathbf{m}_k = \mathbf{m}_{k-1} + \Delta\mathbf{m}_k^*$ is transformed to the body frame using the transformation matrix $\mathbf{C}_{n'}^b = (\mathbf{C}_b^{n'})^T$, to correct the output of the gyroscope and accelerometer measurements. Please, note that the linearized model (7) is used only for finding the equivalent modelling error, while the state vector $\mathbf{x}_{k+1} = \begin{bmatrix}\delta v_N^{n'} & \delta v_N^{n'} & \varphi_N & \varphi_E & \varphi_D\end{bmatrix}^T$ is estimated by using the proposed PKF algorithm as is shown in Fig. 3, after resolving the unknown equivalent modelling error $\mathbf{m}_k$. The state vector contains the estimated errors (misalignment angles and horizontal velocity errors) which are used to provide feedback to the strapped down mechanization to correct the velocity and attitude vectors (see Fig. 2).

5. Experimental results and analysis

The performance of alignment plays a crucial role in the rapidity and convergence of the alignment process. When the azimuth angle is large, the convergence time of the traditional static alignment using the EKF becomes large. In addition, external disturbance can reduce the performance of the EKF method significantly. Hence, in this experiment, the SINS is fixed to a stationary vehicle with its engine running to produce some external disturbances to examine the effectiveness of the proposed alignment method. Moreover, if prior information about the measurement uncertainty matrix $\mathbf{R}$ is not specified accurately, it can cause the divergence of the EKF algorithm. Therefore, the effect of increasing the measurement covariance $\mathbf{R}$ in the convergence performance of the proposed approach and the traditional EKF (10-state filter algorithm) is examined. The convergence time for the EKF is relatively long, while the convergence time for the proposed method is short. Accordingly, we made the tests of the PKF in the post-processing process based on saved real data only to validate the proposed PKF algorithm performance when

the measurement covariance is increased. The PKF is compared to the EKF using the same data; the results demonstrate the effeteness of the PKF in terms of speed and accuracy of the convergence.

5.1. Experiment equipment

In this test, the location of the tested SINS in a stationary car (Fig. 4a) and the hardware consists of an MEMS inertial sensor model AIDS16488A (Fig. 4b), this IMU is fixed to a stationary vehicle body to measure its linear acceleration and its angular rate. The Raspberry pi3 processor is used to store the test data, it communicates with the IMU via an RS-422 serial port. The stored data is used as input to software implementation. Table 1 gives the main specifications of this IMU [17].

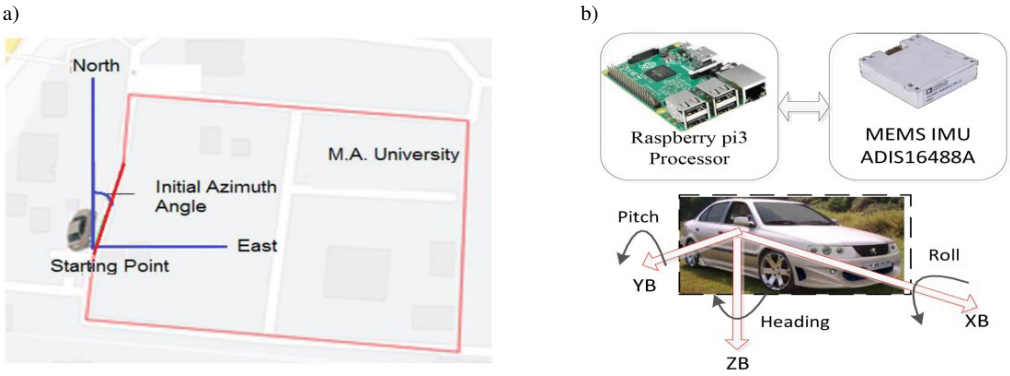


Fig. 4. (a) Test location, (b) Hardware adopted in the experiment.

Table 1. ADIS16488A specifications.

Parameters	Misalignment Axis	Bias Repeatability	Bias Stability	Random Walk	Output Noise	Noise Density
Gyroscopes	± 0.05°	± 0.2°/s	5.1°/hr	0.26°/√hr	0.135°/s	0.0059°/s/√Hz
Accelerometers	± 0.035°	± 16 mg	0.07 mg	0.029 m/s/√hr	1.29 mg	0.063 mg/√Hz

5.2. Results of the alignment test

The alignment process is performed using static SINS data. The test is run as an offline simulation based on saved real data, the test time is 1200 seconds with a sample rate of 125 Hz. To analyse the performance of the designed filter, The PKF is compared to the EKF using the same measured data from the SINS output. In this experiment, a prediction horizon of 10 is used in the PKF.

Table 2 and Table 3 show the results of the influence of increasing the measurement covariance **R** in the convergence performance of the proposed approach and the traditional EKF (10-state filter algorithm). We assume three values for the standard deviation of the measured velocity, these values are $\sigma_R = 10^{-3}$ m/s (This uncertainty is taken from datasheets of sensors), $\sigma_R = 10^{-2}$ m/s and $\sigma_R = 10^{-1}$ m/s. It can be seen that the rate of convergence has increased in the proposed new

method. The convergence performance of the PKF does not vary significantly with increasing the uncertainty of the measurements, but the EKF is not robust enough if measurement noise covariance is increased and its convergence time increases significantly. Accordingly, as the covariance changes more and more, the PKF gains more and more performance relative to the EKF.

Table 2. Results of the alignment experiment for roll and pitch misalignment angles.

TEST No.	PKF				EKF			
	Roll misalignment		Pitch misalignment		Roll misalignment		Pitch misalignment	
	RMS (arc.sec)	STD (arc.sec)	RMS (arc.sec)	STD (arc.min)	RMS (arc.sec)	STD (arc.sec)	RMS (arc.sec)	STD (arc.sec)
1 ($\sigma_R = 10^{-3}$)	0.32	0032	0.3	0.3	0.24	0.23	0.23	0.22
2 ($\sigma_R = 10^{-2}$)	0.035	0.035	0.042	0.029	0.56	0.51	0.73	0.67
3 ($\sigma_R = 10^{-1}$)	0.044	0.005	0.17	0.005	2.35	2.15	1.68	1.56
mean	0.265	0.12	0.17	0.11	1.05	0.96	1.52	1.41

Table 3. Results of the alignment experiment for azimuth misalignment angle.

TEST No.	PKF			EKF		
	RMS (arc.min)	STD (arc.min)	Convergence Time (sec)	RMS (arc.min)	STD (arc.min)	Convergence Time (sec)
1 ($\sigma_R = 10^{-3}$)	0.39	0.14	8	1.12	0.91	150
2 ($\sigma_R = 10^{-2}$)	2.67	0.003	20	11.38	10.42	400
3 ($\sigma_R = 10^{--1}$)	4.087	0.019	50	52.009	47.96	524
mean	2.38	0.054		21.5	19.76	

In the following simulation results of implementing the PKF are shown. The solid lines represent the results of the proposed method and the dashed lines represent the results of the EKF. Fig. 5 illustrates the convergence of both PKF and EKF with respect to three different elements of the measurement noise covariance for initial roll and pitch misalignment angles estimation task. In the case of $\sigma_R = 10^{-1}$ m/s, PKF estimation accuracy of the roll and pitch misalignment angles is approximately $0.044''$ and $0.17''$, respectively, while EKF estimation accuracy of the roll and pitch misalignment angles are approximately $2.35''$ and $1.68''$, respectively. Accordingly, it can be seen in Fig. 5 that the performance of the PKF is better than that of the EKF There is much improvement achieved using a PKF, the new algorithm converges more rapidly and accurately than the EKF filter.

Fig. 6a illustrates the convergence of the both PKF and EKF with respect to three different elements of the measurement noise covariance for initial azimuth misalignment angles estimation task. In the case of $\sigma_R = 10^{-1}$ m/s, the PKF estimation accuracy of the azimuth misalignment angles are approximately $4.087' \pm 0.019'$, while the EKF estimation accuracy of the azimuth misalignment angles is approximately $52' \pm 48'$. However, it is evident from Fig. 6 that the new filter converges rapidly (less than 50 sec) and steadily during its estimation of the azimuth angle, while the EKF filter has a steady state error and a long convergence time (about 524 sec). In the conditions when there are uncertainties in noise covariance, the performance of the EKF is expected to be degraded Simulation results prove this theory.

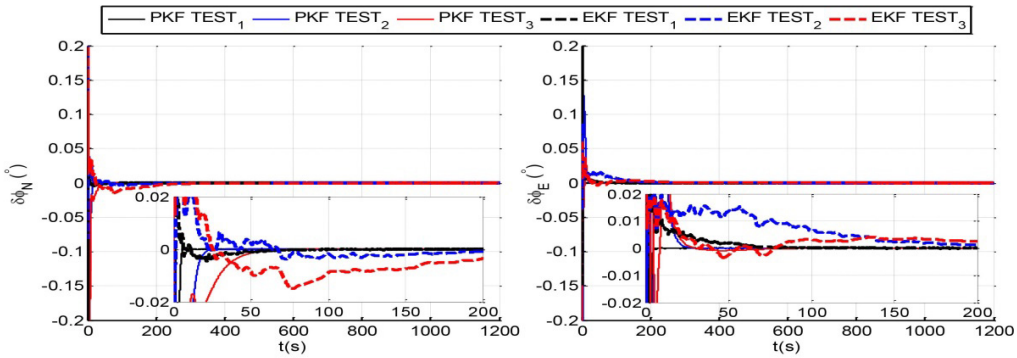


Fig. 5. Comparing the convergence of roll and pitch angles for both PKF and EKF as a function of 3 elements of R.

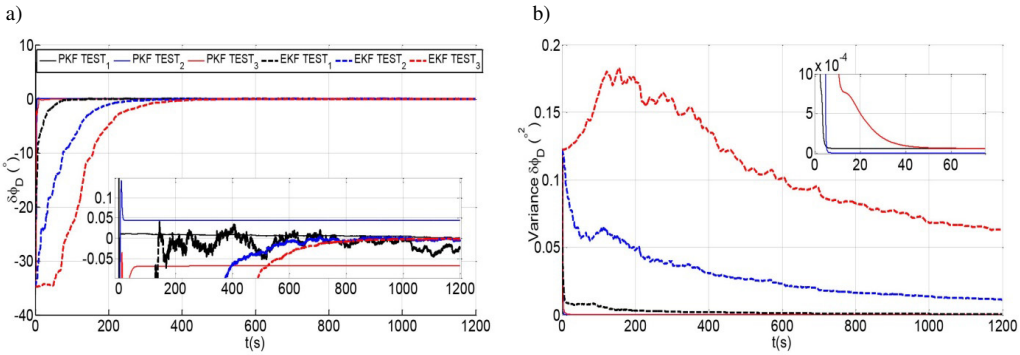


Fig. 6. Results for both PKF and EKF. (a): the convergence of azimuth angle. (b) the errors covariance.

Fig. 6b displays the results of the robustness of both PKF and EKF against uncertainty in measurement noise covariance. The figure shows a significant increase in error covariance of the EKF estimated azimuth misalignment. Therefore, the EKF is not robust with the changes of measurement noise covariance. In conclusion, if prior information about the measurement uncertainty matrix **R** is not specified accurately, it can cause the divergence of the EKF algorithm, while the proposed PKF is robust.

Fig. 7 shows a comparison of results between the PKF and the EKF for the estimated north and east velocity errors for the various changes in the measurement noise covariance. As expected, the new algorithm converges more rapidly than the EKF filter.

Fig. 8 and Fig. 9 show the predicted model errors values of the horizontal accelerometers and the three gyroscopes, these errors are predicted using the proposed filter without augmenting the state variables. Therefore, the observation matrix of the PKF is full rank and all the five state variables are observable, while the ten-state variables of the EKF are not all observables. Therefore, the PKF has an appropriate specification which makes it more suitable for the SINS alignment process without an external aiding.

In this paper, the traditional EKF needs 10-states (the horizontal velocity, misalignment angle and accelerometer and gyroscope bias), while the proposed PKF method has full rank observability. The PKF has 5-states only (the horizontal velocity and misalignment angles), the accelerometer and gyroscope bias is predicted by (17). Therefore, using a fast processor, the

Metrol. Meas. Syst., Vol. 28 (2021), No. 4, pp. 673–691
DOI: 10.24425/mms.2021.137702

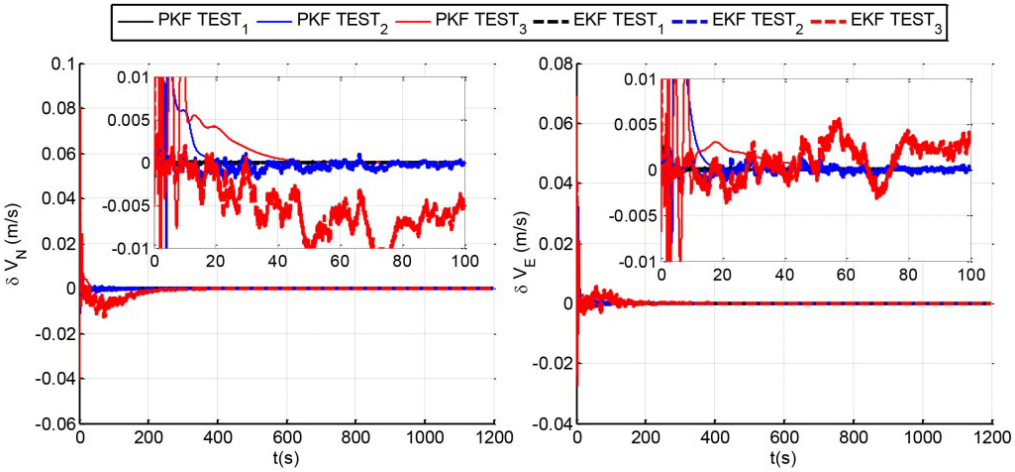


Fig. 7. Comparison of results between the PKF and the EKF for the estimated north and east velocity errors.

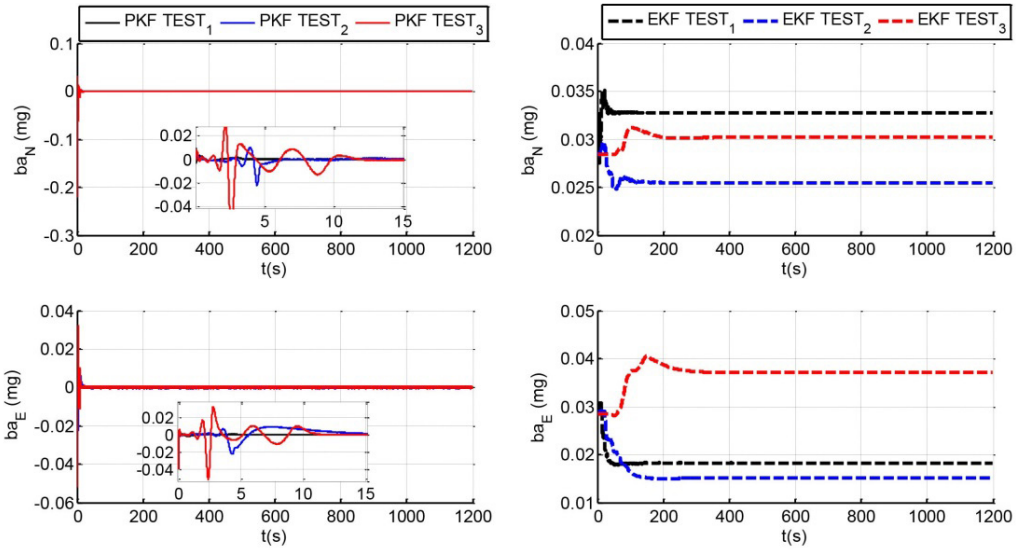


Fig. 8. The predicted model error of the horizontal accelerometers using both PKF and EKF.

PKF can be executed in real-time. On the other hand, the variations of model errors of the sensors are very slow with time. Therefore, the sample time of the optimization process can be increased. Fig. 10a, Fig. 10b and Fig. 10c presents results of the estimated roll, pitch and azimuth misalignment angles, respectively, for real-time PKF static alignment (alignment time equal to 60s and the sampling time equal to 50 ms).

To explain the filter stability, Fig. 11 shows the behaviour of the penalty function values within the prediction horizon; its value does not increase along the system's trajectories via each prediction step. Therefore, the norm of the state is forced to decrease with time and the stability result is reached.

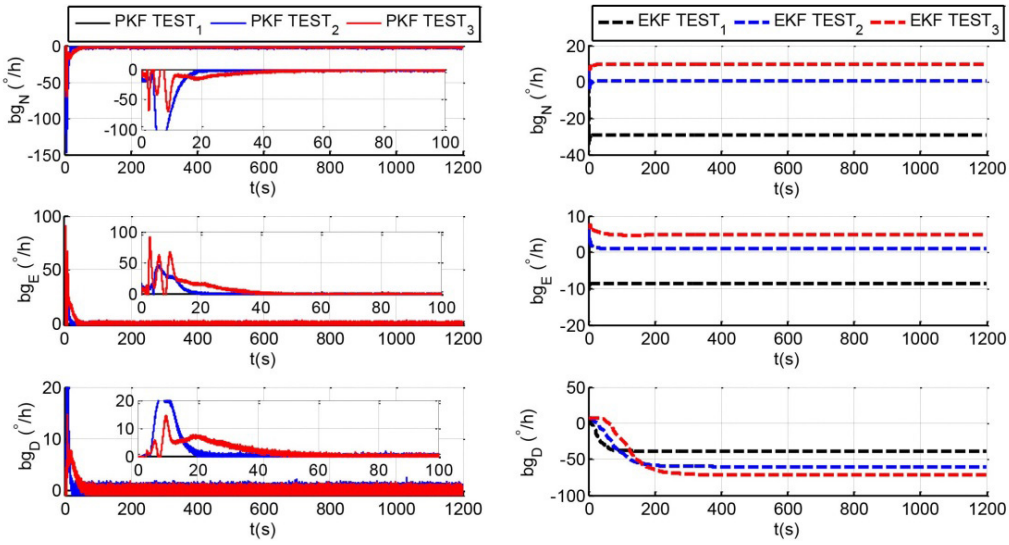


Fig. 9. The predicted model error of the gyroscopes using the PKF and gyroscopes drift using the EKF.

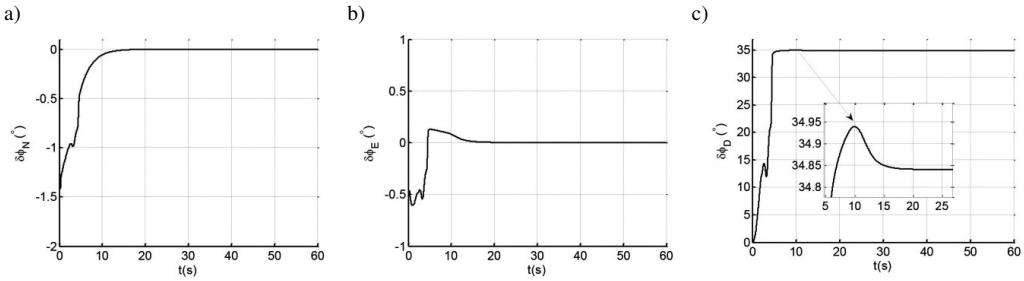


Fig. 10. Real-time PKF, the estimated roll (a), pitch (b) and azimuth (c) misalignment errors.

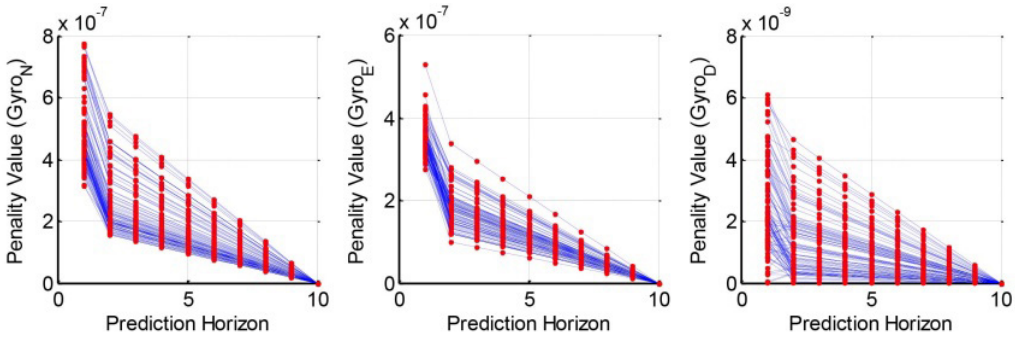


Fig. 11. Values of the penalty function via the prediction horizon ($p = 10$) for 100 sampling time.

6. Conclusions

This paper presents a novel predictive Kalman filter to decrease the static alignment time and to improve the convergence performance during the two phases. First, the PKF algorithm is designed to accurately predict INS errors to adjust the uncertain noise of MEMS-INS. Therefore, the INS errors can be correctly compensated foe. Then an EKF is proposed to fuse the information of MEMS-INS and zero velocity error to estimate the INS misalignment angles and velocity errors to correct the velocity and attitude information of the navigation algorithm using a nonlinear alignment model. To solve the EKF limitations, the proposed PKF approach determines model error sequence which can match a predicted future of observations within a finite horizon. This approach can compensate the predicted equivalent model error without augmenting the system state space. This ensures the stability of the PKF and leads to an accurate and fast SINS alignment process.

A real-life experiment was conducted to check the performance of the proposed algorithm for the static alignment process with a large azimuth angle. The results indicate that the PKF improves the rapidity and the convergence of the whole alignment process and the robustness of the PKF against uncertainty in measurement noise. It can be inferred that the PKF has an appropriate specification which makes it more suitable for the SINS alignment process than the traditional EKF method.

References

[1] Britting, K. R. (1971). *Inertial navigation systems analysis*. Wiley Interscience.

[2] Chang, L., Li, J., & Li, K. (2016). Optimization-based alignment for strapdown inertial navigation system: Comparison and extension. *IEEE Transactions on Aerospace and Electronic Systems*, *52*(4), 1697–1713. https://doi.org/10.1109/TAES.2016.130824

[3] Xue, H., Guo, X., & Zhou, Z. (2016). Parameter identification method for SINS initial alignment under inertial frame. *Mathematical Problems in Engineering*, 2016, 5301242. https://doi.org/10.1155/2016/5301242

[4] Wang, D., Dong, Y., Li, Q., Wu, J., & Wen, Y. (2018). Estimation of small UAV position and attitude with reliable in-flight initial alignment for MEMS inertial sensors. *Metrology and Measurement Systems*, *25*(3), 603–616. https://doi.org/10.24425/123904

[5] Ghanbarpourasl, H. (2020). A new robust quaternion-based initial alignment algorithm for stationary strapdown inertial navigation systems. *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, *234*(12), 1913–1925. https://doi.org/10.1177/0954410020920473

[6] Guo, S., Chang, L., Li, Y., & Sun, Y. (2020). Robust fading cubature Kalman filter and its application in initial alignment of SINS. *Optik*, *202*, 163593. https://doi.org/10.1016/j.ijleo.2019.163593

[7] Zhang, T., Wang, J., Jin, B., & Li, Y. (2019). Application of improved fifth-degree cubature Kalman filter in the nonlinear initial alignment of strapdown inertial navigation system. *Review of Scientific Instruments*, *90*(1), 015111. https://doi.org/10.1063/1.5061790

[8] Xing, H., Chen, Z., Wang, C., Guo, M., & Zhang, R. (2019). Quaternion-based Complementary Filter for Aiding in the Self-Alignment of the MEMS IMU. *2019 IEEE International Symposium on Inertial Sensors and Systems (INERTIAL)*, USA, 1–4. https://doi.org/10.1109/ISISS.2019.8739728

[9] Yang, B., Xu, X., Zhang, T., Sun, J., & Liu, X. (2017). Novel SINS initial alignment method under large misalignment angles and uncertain noise based on nonlinear filter. *Mathematical Problems in Engineering*, 2017, 5917917. https://doi.org/10.1155/2017/5917917

[10] Sun, J., Xu, X., Liu, Y., Zhang, T., & Li, Y. (2015). Initial alignment of large azimuth misalignment angles in SINS based on adaptive UPF. *Sensors*, *15*(9), 21807–21823. https://doi.org/10.3390/s150921807

[11] Han, H., Wang, J., & Du, M. (2017). A fast SINS initial alignment method based on RTS forward and backward resolution. *Journal of Sensors*, 2017, 7161858. https://doi.org/10.1155/2017/7161858

[12] Kaygısız, B. H., & Şen, B. (2015). In-motion alignment of a low-cost GPS/INS under large heading error. *The Journal of Navigation*, *68*(2), 355–366. https://doi.org/10.1017/S0373463314000629

[13] Xia, X., & Sun, Q. (2018). Initial alignment algorithm based on the DMCS method in single-axis RSINS with large azimuth misalignment angles for submarines. *Sensors*, *18*(7), 1807–2123. https://doi.org/10.3390/s18072123

[14] Li, J., Gao, W., Zhang, Y., & Wang, Z. (2018). Gradient Descent Optimization-Based Self-Alignment Method for Stationary SINS. *IEEE Transactions on Instrumentation and Measurement*, *68*(9), 3278–3286. https://doi.org/10.1109/TIM.2018.2878071

[15] Camacho, E. F., Ramírez, D. R., Limón, D., De La Peña, D. M., & Alamo, T. (2010). Model predictive control techniques for hybrid systems. *Annual Reviews in Control*, *34*(1), 21–31. https://doi.org/10.1016/j.arcontrol.2010.02.002

[16] Titterton, D., Weston, J. L., & Weston, J. (2004). *Strapdown inertial navigation technology*. IET. https://doi.org/10.1049/PBRA017E

[17] Analog Devices. (2018). *Tactical Grade Ten Degrees of Freedom Inertial Sensor – ADIS16488A*. [Datasheet, Rev. F]. https://www.analog.com/media/en/technical-documentation/data-sheets/ADIS16488A.pdf



Hasssan Majed Alhassan received his B.Sc. degree in System Electronic Engineering from the Higher Institute of Applied Sciences and Technology, Damascus, Syria, in 1997 and his M.Sc. degree in Control and Navigation Engineering from Malek Ashtar University of Technology, Tehran, Iran, in 2010. He is currently working on his Ph.D. thesis in Integrated Navigation Systems at the Department of Electrical and Computer Engineering of Malek Ashtar University of Technology, Tehran, Iran. His primary research interests include the land-vehicle navigation, theory and application of inertial sensors, integrated navigation and estimation methods.



Nemat Allah Ghahremani received his M.Sc. degree in Electrical Engineering, from Ferdowsi University, Mashad, Iran, in 1991 and his Ph.D. degree in Control Engineering from Amirkabir University, Tehran, Iran in 2008. He is currently a researcher and Doctoral Supervisor of the Department of Electrical and Computer Engineering of Malek Ashtar University of Technology, Tehran, Iran. He has published several technical papers and proceedings articles. His research includes system optimization, model based predictive control, predictive filters, industrial automation and navigation systems.

APPENDIX A: The nonlinear attitude equation derivation

In (19), the time rate of change of direction cosine matrix (DCM) $\mathbf{C}_b^n$ is linked to angular velocity matrix $\mathbf{\Omega}_{nb}^b$ via [1]:

$$\dot{\mathbf{C}}_b^n = \mathbf{C}_b^n \mathbf{\Omega}_{nb}^b = \mathbf{C}_b^n (\mathbf{\Omega}_{ib}^b - \mathbf{\Omega}_{in}^b), \tag{A1}$$

where matrix $\mathbf{\Omega}_{nb}^b$ is the skew-symmetric form of $\boldsymbol{\omega}_{nb}^b$; the body rate relative to navigation frame resolved in the body frame; vector $\boldsymbol{\omega}_{nb}^b$ is $\boldsymbol{\omega}_{nb}^b = \boldsymbol{\omega}_{ib}^b - \mathbf{C}_n^b \boldsymbol{\omega}_{in}^n$. Also, matrix $\mathbf{\Omega}_{ib}^b$ is the skew-symmetric form of $\boldsymbol{\omega}_{ib}^b$, which is the true angular rate of the body frame relative to the inertial frame, resolved in the body frame. Matrix $\mathbf{\Omega}_{in}^b$ represents the skew-symmetric matrix of true

angular rate $\boldsymbol{\omega}_{in}^{b}$ of the navigation frame relative to the inertial frame coordinated in the body frame. The computed version of (A1) is [16]:

$$\dot{\hat{\mathbf{C}}}_{b}^{n} = \hat{\mathbf{C}}_{b}^{n}(\hat{\boldsymbol{\Omega}}_{ib}^{b} - \hat{\boldsymbol{\Omega}}_{in}^{b}). \tag{A2}$$

In this paper, the error analysis uses perturbation method to linearize the nonlinear system differential equation (A2). Perturbation analysis involves the substitution $\hat{\mathbf{C}}_{b}^{n} = \mathbf{C}_{b}^{n'} = \mathbf{C}_{b}^{n} + \delta\mathbf{C}_{b}^{n}$, where matrix $\hat{\mathbf{C}}_{b}^{n}$ is the computed DCM matrix, denoted as $\mathbf{C}_{b}^{n'}$, matrix $\mathbf{C}_{b}^{n}$ is the true DCM and $\delta\mathbf{C}_{b}^{n}$ is the computed DCM matrix error. Therefore,

$$\delta\mathbf{C}_{b}^{n} = \mathbf{C}_{b}^{n'} - \mathbf{C}_{b}^{n} \tag{A3}$$

or

$$\delta\mathbf{C}_{b}^{n} = (\mathbf{I} - \mathbf{C}_{n'}^{n})\mathbf{C}_{b}^{n'}. \tag{A4}$$

Since $\dot{\mathbf{C}}_{b}^{n} = \mathbf{C}_{b}^{n}\boldsymbol{\Omega}_{nb}^{b} = \mathbf{C}_{b}^{n}\boldsymbol{\Omega}_{ib}^{b} - \boldsymbol{\Omega}_{in}^{n}\mathbf{C}_{b}^{n}$ [16], then $\dot{\mathbf{C}}_{b}^{n'} = \mathbf{C}_{b}^{n'}\hat{\boldsymbol{\Omega}}_{nb}^{b} = \mathbf{C}_{b}^{n'}\hat{\boldsymbol{\Omega}}_{ib}^{b} - \hat{\boldsymbol{\Omega}}_{in}^{n}\mathbf{C}_{b}^{n'}$, the derivative of (A3) gives the following formula:

$$\delta\dot{\mathbf{C}}_{b}^{n} = \dot{\mathbf{C}}_{b}^{n'} - \dot{\mathbf{C}}_{b}^{n} = \mathbf{C}_{b}^{n'}\hat{\boldsymbol{\Omega}}_{ib}^{b} - \hat{\boldsymbol{\Omega}}_{in}^{n}\mathbf{C}_{b}^{n'} - \mathbf{C}_{b}^{n}\boldsymbol{\Omega}_{ib}^{b} + \boldsymbol{\Omega}_{in}^{n}\mathbf{C}_{b}^{n}. \tag{A5}$$

Similarly, the derivative of (A4) gives the following formula:

$$\delta\dot{\mathbf{C}}_{b}^{n} = -\dot{\mathbf{C}}_{n'}^{n}\mathbf{C}_{b}^{n'} + (\mathbf{I} - \mathbf{C}_{n'}^{n})\dot{\mathbf{C}}_{b}^{n'} = -\dot{\mathbf{C}}_{n'}^{n}\mathbf{C}_{b}^{n'} + \left(\mathbf{I} - \mathbf{C}_{n'}^{n}\right)\left(\mathbf{C}_{b}^{n'}\hat{\boldsymbol{\Omega}}_{ib}^{b} - \hat{\boldsymbol{\Omega}}_{in}^{n}\mathbf{C}_{b}^{n'}\right). \tag{A6}$$

The perturbation method gives $\hat{\boldsymbol{\Omega}}_{ib}^{b} = \boldsymbol{\Omega}_{ib}^{b} + \delta\boldsymbol{\Omega}_{ib}^{b}$, and the two equations (A5) and (A6) are equivalent Thus, the following equation is obtained:

$$\mathbf{C}_{n'}^{n}\boldsymbol{\Omega}_{nn'}^{n'}\mathbf{C}_{b}^{n'} + \mathbf{C}_{b}^{n}\delta\boldsymbol{\Omega}_{ib}^{b} + \boldsymbol{\Omega}_{in}^{n}\mathbf{C}_{b}^{n} - \mathbf{C}_{n'}^{n}\hat{\boldsymbol{\Omega}}_{in}^{n}\mathbf{C}_{b}^{n'} = 0. \tag{A7}$$

Right multiplying $\mathbf{C}_{n'}^{b}$ and left multiplying $\mathbf{C}_{n}^{n'}$ to the above equation gives:

$$\boldsymbol{\Omega}_{nn'}^{n'} + \mathbf{C}_{b}^{n'}\delta\boldsymbol{\Omega}_{ib}^{b}\mathbf{C}_{n'}^{b} + \mathbf{C}_{n}^{n'}\boldsymbol{\Omega}_{in}^{n}\mathbf{C}_{n'}^{n} - \hat{\boldsymbol{\Omega}}_{in}^{n} = 0. \tag{A8}$$

Because the skew-symmetric matrices transform under the similarity transformation [1], then, $\boldsymbol{\Omega}_{in}^{n'} = \mathbf{C}_{n}^{n'}\boldsymbol{\Omega}_{in}^{n}\mathbf{C}_{n'}^{n}$ and $\delta\boldsymbol{\Omega}_{ib}^{n'} = \mathbf{C}_{b}^{n'}\delta\boldsymbol{\Omega}_{ib}^{b}\mathbf{C}_{n'}^{b}$. Therefore (A8) is rewritten as:

$$\boldsymbol{\Omega}_{nn'}^{n'} + \delta\boldsymbol{\Omega}_{ib}^{n'} + \boldsymbol{\Omega}_{in}^{n'} - \boldsymbol{\Omega}_{in}^{n} - \delta\boldsymbol{\Omega}_{in}^{n} = 0 \tag{A9}$$

The matrix form of (A9) is converted to vectors form as follows:

$$\boldsymbol{\omega}_{nn'}^{n'} = \boldsymbol{\omega}_{in}^{n} - \boldsymbol{\omega}_{in}^{n'} + \delta\boldsymbol{\omega}_{in}^{n} - \delta\boldsymbol{\omega}_{ib}^{n'} = \left(\mathbf{I} - \mathbf{C}_{n}^{n'}\right)\boldsymbol{\omega}_{in}^{n} + \delta\boldsymbol{\omega}_{in}^{n} - \mathbf{C}_{b}^{n'}\delta\boldsymbol{\omega}_{ib}^{b}. \tag{A10}$$

Vector $\boldsymbol{\varphi} = [\varphi_N \quad \varphi_E \quad \varphi_D]^T$ denotes the misalignment angles between the computed navigation frame and the real navigation frame. If horizontal misalignment angles φ_N and φ_E are small and azimuth misalignment angle φ_D is large, the SINS attitude error rate changes is modelled as follows:

$$\dot{\boldsymbol{\varphi}} \approx \boldsymbol{\omega}_{nn'}^{n'} = (\mathbf{I} - \mathbf{C}_{n}^{n'})\boldsymbol{\omega}_{in}^{n} + \delta\boldsymbol{\omega}_{in}^{n} - \mathbf{C}_{b}^{n'}\delta\boldsymbol{\omega}_{ib}^{b}, \tag{A11}$$

where vectors $\dot{\boldsymbol{\varphi}}$ and $\boldsymbol{\omega}_{nn'}^{n'}$ denote the angular rate between the computed navigation frame and the real navigation frame, matrix $\mathbf{I} \in \mathbb{R}^3$ is the unity matrix, matrix $\mathbf{C}_{n}^{n'} = \left(\mathbf{C}_{n'}^{n}\right)^T$ takes the form of (23), the angular rate of navigation frame $\boldsymbol{\omega}_{in}^{n}$ and its perturbation error $\delta\boldsymbol{\omega}_{in}^{n}$ are given as presented in (25), matrix $\mathbf{C}_{b}^{n'}$ is calculated as follows $\mathbf{C}_{b}^{n'} = \mathbf{C}_{n}^{n'}\mathbf{C}_{b}^{n}$, vector $\delta\boldsymbol{\omega}_{ib}^{b}$ is the gyroscopes model error.